



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/750,455	12/31/2003	Michael Swafford	50037.0237US01	4974

27488 7590 03/21/2007
MERCHANT & GOULD (MICROSOFT)
P.O. BOX 2903
MINNEAPOLIS, MN 55402-0903

EXAMINER

RUTZ, JARED IAN

ART UNIT	PAPER NUMBER
----------	--------------

2187

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	03/21/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.		Applicant(s)	
	10/750,455		SWAFFORD ET AL.	
	Examiner		Art Unit	
	Jared I. Rutz		2187	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 27 December 2006.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-7, 9-18, 20-29 and 31-33 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-7, 9-18, 20-29 and 31-33 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. Claims 1-7, 9-18, 20-29, and 31-33, as amended on 12/27/2006, are pending in the instant application. Independent claims 1, 11, and 21 have been amended to incorporate limitations indicated allowable in the Office action of 6/27/2006. However, this Office action contains new grounds of rejection, not necessitated by the amendment to the claims. Accordingly, this Office action is made non-final. The Examiner apologizes for any inconvenience caused by the withdrawal of the indication of allowable subject matter.

Claim Rejections - 35 USC § 112

2. The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

3. **Claims 11-18, 20 and 32** are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention. Claims 11-18, 20 and 32 are directed to "*A computer-readable medium arranged for storing computer executable components*". Page 4 line 17 through page 5 line 12 discuss computer readable media, which is shown to include both storage media and communication media. The Examiner is not aware of a portion of the specification that teaches which types of

Art Unit: 2187

computer readable media are "arranged for storing computer readable media" as recited in claim 1.

Claim Rejections - 35 USC § 101

4. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

5. **Claims 11-18, 20 and 32** are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Claims 11-18, 20 and 32 are directed to "*A computer-readable medium arranged for storing computer executable components*". Page 4 line 17 through page 5 line 12 discuss computer readable media, which is shown to include both storage media and communication media.

Communication media is shown to be embodied by computer readable instructions in a modulated data signal, such as a carrier wave. Electromagnetic waves are not considered statutory subject matter as they are not a process, machine, manufacture, or combination of matter. Applicant may overcome this rejection to amend claims 11-18 and 32 to be directed to a computer storage medium having computer executable components.

Claim Rejections - 35 USC § 102

6. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

(a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

7. **Claims 11-18, 20, and 32** are rejected under 35 U.S.C. 102(b) as being anticipated by Voigt (US 6,092,168).

8. **Claim 11** recites "*A computer-readable medium arranged for storing computer-executable components for overwrite detection within an allocable memory block*". As Voigt discloses a memory, item 42 of figure 2, which stores computer-executable components, operating system 44, free space collection program 50, Voigt discloses A computer-readable medium arranged for storing computer executable components.

9. **Claims 11-18, 20, and 32** further recite computer executable components, but do not positively recite that the claimed computer-readable medium stores the recited components. Accordingly, they are anticipated by the disclosure of Voigt.

10. **Claims 1-7, 9, 11-18, and 31-32** are rejected under 35 U.S.C. 102(a) as being anticipated by Robertson et al. (Run-time Detection of Heap-based Overflows).

11. **Claim 1** is taught by Robertson as:

a. *A method for providing overwrite detection for an allocable memory block comprising: receiving a request for performing one of requesting the allocable*

memory block, requesting the size of the allocable memory block, and freeing the allocable memory block. Page 53, column 1, lines 3-8 discuss malloc and free, C language functions for requesting allocation of a block of memory and freeing an allocated block of memory.

b. *Generating an overwrite detection pattern for the allocable memory block.* Page 55, column 1, lines 15-18 shows that the canary is a checksum covering the memory location of the chunk and the chunks size field seeded with a global value `__heap_magic`.

c. *Storing the overwrite detection pattern in the allocable memory block.* Page 55, column 1, lines 15-18 show that newly allocated chunks to be returned from malloc have their canary initialized.

d. *Checking the overwrite detection pattern.* Page 55, column 1, lines 23-26 shows that when the chunk is returned by a call to free, the chunk's canary is checked against the checksum calculation performed when the chunk was allocated.

e. *And forcing an access violation if the overwrite detection pattern has been modified.* Page 55, column 1, lines 26-30 shows that if the stored value does not match the current calculation, a corruption of the management information is assumed and an alert is raised.

12. **Claim 2** is taught by Robertson as:

- f. *The method of claim 1, further comprising examining the heap to determine whether the overwrite detection pattern has been overwritten.* Page 55, column 1, lines 23-26 shows that when the chunk is returned by a call to free, the chunk's canary is checked against the checksum calculation performed when the chunk was allocated.
13. **Claim 3** is taught by Robertson as:
- g. *The method of claim 1, further comprising performing a checksum on the allocable memory block and storing the results of the checksum within the allocable memory block.* Page 54, column 2, lines 6-12 shows that the canary is a checksum and prepended to the chunk structure.
14. **Claim 4** is taught by Robertson as:
- h. *The method of claim 3, further comprising examining the results of the checksum to determine the presence of memory errors.* Page 55, column 1, line 54 through page 55 column 2, line 8 shows that a double free is detected because the checksum is checked and does not match the expected value.
15. **Claim 5** is taught by Robertson as:
- i. *The method of claim 1, wherein the overwrite detection pattern is written at the end of the allocable memory block.* Page 54, column 2, lines 6-12 shows that the canary is a checksum and prepended to the chunk structure.

Prepending the canary value places it at the top end of the chunk as shown in figure 3.

16. **Claim 6** is taught by Robertson as:

j. *The method of claim 1, wherein a logical function of the elements within the overwrite detection pattern provides a predetermined result. Logically ANDing the canary value with a 0 will always produce 0, a predetermined result. Page 55, column 1, lines 23-26 shows that when the chunk is returned by a call to free, the chunk's canary is checked against the checksum calculation performed when the chunk was allocated.*

17. **Claim 7** is taught by Robertson as:

k. *The method of claim 1, wherein the overwrite detection pattern is written within an area of the allocable memory block that is used for alignment purposes. Page 54 lines 10-13 shows that __pad0 is also added to the header because dlmalloc requires the size of a header of a used chunk to be a power of two.*

18. **Claim 9** is taught by Robertson as:

l. *The method of claim 1, further comprising storing a heap index for the allocable memory block within the allocable memory block, wherein the heap index points to one of a plurality of heaps. Page 53 lines 33-37 shows that unallocated chunks store pointers to other unallocated chunks in the same bin.*

19. **Claim 11** is taught by Robertson as:

- m. *A computer-readable medium arranged for storing computer-executable components for overwrite detection within an allocable memory block. The second paragraph of the abstract shows that the detection scheme proposed in the paper has been implemented as a patch to the GNU Lib C. Page 57, column 2, lines 7-8 shows that the library modifications can be downloaded and installed as a patch for glibc.*
- n. *Comprising: a first component that is arranged to receive a request for performing one of requesting the allocable memory block, requesting the size of the allocable memory block, and freeing the allocable memory block. Page 53, column 1, lines 3-8 discuss malloc and free, C language functions for requesting allocation of a block of memory and freeing an allocated block of memory.*
- o. *A second component that is arranged to generate an overwrite detection pattern for the allocable memory block. Page 55, column 1, lines 15-18 shows that the canary is a checksum covering the memory location of the chunk and the chunks size field seeded with a global value __heap_magic.*
- p. *A third component that is arranged to store the overwrite detection pattern in the allocable memory block. Page 55, column 1, lines 15-18 show that newly allocated chunks to be returned from malloc have their canary initialized.*
- q. *And a fourth component that is arranged to store a heap index for the allocable memory block within the allocable memory block, wherein the heap*

index points to one of a plurality of heaps. Page 53 lines 33-37 shows that unallocated chunks store pointers to other unallocated chunks in the same bin.

20. **Claim 12** is taught by Robertson as:

r. *The computer-readable medium of claim 11, further comprising an examination component that is arranged to examine the heap to determine whether the overwrite detection pattern has been overwritten. Page 55, column 1, lines 23-26 shows that when the chunk is returned by a call to free, the chunk's canary is checked against the checksum calculation performed when the chunk was allocated.*

21. **Claim 13** is taught by Robertson as:

s. *The computer-readable medium of claim 11, further comprising a checksum component that is arranged to perform a checksum on the allocable memory block and storing the results of the checksum within the allocable memory block. Page 54, column 2, lines 6-12 shows that the canary is a checksum and prepended to the chunk structure.*

22. **Claim 14** is taught by Robertson as:

t. *The computer-readable medium of Claim 13, further comprising a checksum examination component that is arranged to examine the results of the checksum to determine the presence of memory errors. Page 55, column 1, line*

54 through page 55 column 2, line 8 shows that a double free is detected because the checksum is checked and does not match the expected value.

23. **Claim 15** is taught by Robertson as:

u. *The computer-readable medium of claim 11, wherein the overwrite detection pattern is written at the end of the allocable memory block. Page 54, column 2, lines 6-12 shows that the canary is a checksum and prepended to the chunk structure. Prepending the canary value places it at the top end of the chunk as shown in figure 3.*

24. **Claim 16** is taught by Robertson as:

v. *The computer-readable medium of claim 11, wherein a logical function of the elements within the overwrite detection pattern provides a predetermined result. Logically ANDing the canary value with a 0 will always produce 0, a predetermined result. Page 55, column 1, lines 23-26 shows that when the chunk is returned by a call to free, the chunk's canary is checked against the checksum calculation performed when the chunk was allocated.*

25. **Claim 17** is taught by Robertson as:

w. *The computer-readable medium of claim 11, wherein the overwrite detection pattern is written within an area of the allocable memory block that is used for alignment purposes. Page 54 lines 10-13 shows that __pad0 is also*

added to the header because `dlmalloc` requires the size of a header of a used chunk to be a power of two.

26. **Claim 18** is taught by Robertson as:

x. *The computer-readable medium of Claim 11, wherein the overwrite detection pattern is checked and an access violation is forced if the overwrite detection pattern has been modified. Page 55, column 1, lines 23-30 shows that the stored value is checked and if the stored value does not match the current calculation, a corruption of the management information is assumed and an alert is raised.*

27. **Claim 31** is taught by Robertson as:

y. *The method of Claim 1, wherein the overwrite detection pattern is checked when the allocable memory block is passed back to the operating system. Page 55, column 1, lines 23-30 shows that the stored value is checked when the chunk is returned to the heap management through a call to free.*

28. **Claim 32** is taught by Robertson as:

z. *The computer-readable medium of claim 18, wherein the overwrite detection pattern is checked when the allocable memory block is passed back to the operating system. Page 55, column 1, lines 23-30 shows that the stored*

value is checked when the chunk is returned to the heap management through a call to free.

Claim Rejections - 35 USC § 103

29. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

30. **Claims 10, 20, 21-29, and 33** are rejected under 35 U.S.C. 103(a) as being unpatentable over Robertson et al. (cited supra) in view of Gupta (Tasks and Task Management).

31. **Claim 10** is taught by Robertson as shown supra with respect to claim 1.

32. Robertson does not expressly disclose the use of a timestamp stored within the allocable memory block which indicates the time the block was requested or freed.

33. With respect to claim 10, Gupta teaches:

aa. *The method of Claim 1, further comprising storing a timestamp within the allocable memory block, wherein the timestamp indicates the time when one of requesting and freeing the allocable memory block is performed. Slide 33 on page 17, titled Other Heap Issues, teaches saving information with each block of*

memory to help characterize memory usage and performance. Lines 10-13 teach the use of a timestamp to see how long a memory block has been allocated.

34. Robertson and Gupta are analogous art because they are from the same field of endeavor, computer memory management.

35. At the time of the invention it would have been obvious to a person of ordinary skill in the art to include a timestamp indicating the time a memory block was requested in the allocated memory block.

36. The motivation for doing so would have been monitor how long a memory block has been allocated, which can identify the presence of possible memory leaks, Gupta page 17 lines 10-16.

37. Therefore, it would have been obvious to combine Gupta with Robertson for the benefit of detecting possible memory leaks to obtain the invention as specified in **claims 10, 20, 21-29, and 33**.

38. **Claim 20** is taught by Gupta as:

bb. *The computer-readable medium of Claim 11, further comprising a timestamp component that is arranged to store a timestamp within the allocable memory block, wherein the timestamp indicates the time when one of requesting and freeing the allocable memory block is performed.*

39. **Claim 21** is taught the combination of Robertson and Gupta as:

cc. *A system for overwrite detection in an allocable memory block, comprising: a computer memory that comprises a heap in which an allocable memory block can be allocated and freed.* Gupta at page 16 teaches the use of heaps in memory for memory allocation.

dd. *A memory allocator that is arranged to receive a request for performing one of requesting the allocable memory block, requesting the size of the allocable memory block, and freeing the allocable memory block.* Robertson page 53, column 1, lines 3-8 discuss malloc and free, C language functions for requesting allocation of a block of memory and freeing an allocated block of memory.

ee. *A pattern generator that is arranged to generate an overwrite detection pattern for the allocable memory block.* Page 55, column 1, lines 15-18 shows that the canary is a checksum covering the memory location of the chunk and the chunks size field seeded with a global value `__heap_magic`.

ff. *And a memory timestamp system that is arranged to store a timestamp within the allocable memory block, wherein the timestamp indicates the time when one of requesting and freeing the allocable memory block is performed.*

Gupta slide 33 on page 17, titled Other Heap Issues, teaches saving information with each block of memory to help characterize memory usage and performance. Lines 10-13 teach the use of a timestamp to see how long a memory block has been allocated.

40. **Claim 22** is taught by Robertson as:

gg. *The system of Claim 21, further comprising a memory verification system that is arranged to examine a heap to determine whether the overwrite detection pattern has been overwritten. Page 55, column 1, lines 23-26 shows that when the chunk is returned by a call to free, the chunk's canary is checked against the checksum calculation performed when the chunk was allocated.*

41. **Claim 23** is taught by Robertson as:

hh. *The system of Claim 21, further comprising a memory verification system that is arranged to perform a checksum on the allocable memory block and storing the results of the checksum within the allocable memory block. Page 54, column 2, lines 6-12 shows that the canary is a checksum and prepended to the chunk structure.*

42. **Claim 24** is taught by Robertson as:

ii. *The system of claim 23, further comprising a memory verification system that is arranged to examine the results of the checksum to determine the presence of memory errors. Page 55, column 1, line 54 through page 55 column 2, line 8 shows that a double free is detected because the checksum is checked and does not match the expected value.*

43. **Claim 25** is taught by Robertson as:

jj. *The system of Claim 21, wherein the overwrite detection pattern is written at the end of the allocable memory block.* Page 54, column 2, lines 6-12 shows that the canary is a checksum and prepended to the chunk structure. Prepending the canary value places it at the top end of the chunk as shown in figure 3.

44. **Claim 26** is taught by Robertson as:

kk. *The system of Claim 21, wherein a logical function of the elements within the overwrite detection pattern provides a predetermined result.* Logically ANDing the canary value with a 0 will always produce 0, a predetermined result. Page 55, column 1, lines 23-26 shows that when the chunk is returned by a call to free, the chunk's canary is checked against the checksum calculation performed when the chunk was allocated.

45. **Claim 27** is taught by Robertson as:

ll. *The system of Claim 21, wherein the memory overwrite detection pattern is written within an area of the allocable memory block that is used for alignment purposes.* Page 54 lines 10-13 shows that __pad0 is also added to the header because dlmalloc requires the size of a header of a used chunk to be a power of two.

46. **Claim 28** is taught by Robertson as:

mm. *The system of claim 21, wherein the overwrite detection pattern is checked and an access violation is forced if the overwrite detection pattern has been modified.* Page 55, column 1, lines 23-30 shows that the stored value is checked and if the stored value does not match the current calculation, a corruption of the management information is assumed and an alert is raised.

47. **Claim 29** is taught by Robertson as:

nn. *The system of Claim 21, further comprising a memory indexing system that is arranged to store a heap index for the allocable memory block within the allocable memory block, wherein the heap index points to one of a plurality of heaps.* Page 53 lines 33-37 shows that unallocated chunks store pointers to other unallocated chunks in the same bin.

48. **Claim 33** is taught by Robertson as:

oo. *The system of claim 28, wherein the overwrite detection pattern is checked when the allocable memory block is passed back to the operating system.* Page 55, column 1, lines 23-30 shows that the stored value is checked when the chunk is returned to the heap management through a call to free.

Response to Arguments

49. In the second paragraph beginning on page 8 of Applicant's Arguments submitted 12/27/2006, with respect to the rejection of claims 11-18, 20, and 32 under 35

USC 101, applicant states "*Applicants have amended claim 11 to direct the claim toward statutory subject matter in including 'a computer-readable medium arranged for storing computer-executable components.'*" The Examiner respectfully disagrees. As noted supra, there is no teaching in the specification as originally filed as to what a computer-readable medium arranged for storing computer-executable components comprises. Computer readable medium are taught in the specification as originally filed to include communication media. Accordingly, the amendment to claim 11 is not sufficient to overcome the rejection of claims 11-18, 20, and 32 under 35 USC 101.

50. Additionally, the Examiner respectfully notes that claim 11, as currently amended, does not require the storing of the recited computer-executable components, merely requiring that the computer readable medium is arranged to store the recited components.

Conclusion

51. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

52. Etoh et al. (US 2001/0013094) teaches a method of using guard variables to allow detection of stack overwrite errors.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jared I. Rutz whose telephone number is (571) 272-5535. The examiner can normally be reached on M-F 8:00 AM - 4:00 PM.


If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Donald Sparks can be reached on (571) 272-4201. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Jared I Rutz
Examiner
Art Unit 2187

jir

dit


Brian R. Peigh
Primary Examiner
3/19/07